

REMARKS

The examiner removed the indication of allowable subject matter of claims 23-24 in view of a newly discovered reference to Venkatraman (U.S. Patent No. 5,923,849).

35 U.S.C § 102

The examiner rejected Claims 1-3, 5, 7-16, and 18-22 under 35 U.S.C. 102 (e) as being anticipated by Ontiveros et al. (20020107953).

The examiner stated:

As per claim 1, Ontiveros discloses a system, comprising:
a plurality of collector devices that are disposed to collect connection information to identify host connection pairs from packets that are sent between nodes on a network (paragraph [0024])
an aggregator device that receives the connection information from the plurality of collector devices (paragraph [0037]), and which produces a connection table that maps each node on the network to a record that stores information about packet traffic to or from the node (paragraph [0040]), with the aggregator device further comprising:
a process executed on the aggregator device to detect anomalies in connection patterns (paragraphs [0008] and [0024])
a process executed on the aggregator device to aggregate detected anomalies into the network events (paragraph [0026], Anomaly Detection System) with the anomalies that are detected including denial of service attack anomalies and scanning attack anomalies (paragraphs [0003] and [0024]).

Claim 1 includes the features of ... an aggregator device that receives the connection information from the plurality of collector devices, and which produces a connection table that maps each node on the network to a record that stores information about packet traffic to or from the node, ... the aggregator device to detect anomalies in connection patterns and ... to aggregate detected anomalies into the network events with the anomalies that are detected including denial of service attack anomalies and scanning attack anomalies.

The examiner relies on Ontiveros [0040] for the features of the connection table. Ontiveros [0040] is reproduced below:

[0040] With respect more specifically to the "hit-count" table, each time a data packet is received, a preferred algorithm as described herein creates a new reference index (if one does not already exist) or increments the existing reference (i.e., counting packets) . For example, as shown at 100 in FIG. 2, the packet daemon

identifies the packet source address qw1232ewr23 and at 102 creates a memory reference (memref) for that source address. At 104 the packet daemon identifies the source address of the next data packet traversing the port being monitored by the packet daemon, in FIG. 2, the source address being mg32ewr009. At 106 another memref is created for this source address. Therefore, at 104 each of the memrefs are equal to 1, representing that one data packet from each of the sources identified has traversed the data port of interest. At 108, another packet from source address gw1232ewr23 is identified, and as shown at 110, the corresponding memref for that address is incremented. So, for example the threshold data packet value is 1000 for the sample time (e.g., 10 milliseconds), and source address qw1232ewr23 exceeds the threshold in this period (e.g., memref qw1232ewr23=1001), then access to the port being monitored will be denied to packets from that source. It should be noted that the source may be transmitting from either outside or inside the network.

The examiner does not specifically point out what in [0040] corresponds to the connection table. Applicant presumes that the examiner uses the teachings of the hit-count table. The hit-count table is as its name implies, counts the number of times that a pair of source and destination addresses is detected. Ontiveros states:

A "hit-count" table is preferably created in memory to count the number of times a particular pair of source and destination IP addresses is detected. Entries are stored using a hash table, keyed by the source and destination addresses. In operation, if the "hit" count exceeds a configurable threshold, all traffic between the source and destination endpoints is disabled for a configurable lockout period. When the lockout period ends, traffic between the endpoints is re-enabled. The IDS of the monitoring system 50 preferably generates a system log message when a lockout period begins or ends.

Thus, while the hit-count table tracks the number of times a particular pair of source and destination IP addresses is detected, the hit count table neither describes nor suggests the feature of "a connection table that maps each node on the network to a record that stores information about packet traffic to the node and from the node,"

By contrast, the connection table is a data structure that maps each host (e.g., identified by IP address) to a "host object" that stores information about all traffic to or from that host. In one implementation of the table, source address is one dimension, destination is a second dimension and time is a third dimension. The time dimension allows a current record and historical records to be maintained.

Claim 1 further includes the features of a process executed on the aggregator device to detect anomalies in connection patterns. The examiner contends that (paragraphs [0008] and [0024]) of Ontiveros disclose this feature.

[0008] The invention is preferably provided as an intrusion detection system (IDS) using a packet daemon that captures, sorts, and catalogs network traffic on a packet-by-packet basis. The packets are preferably captured for inspection by an interface, for example, by using available libpcap libraries. These libraries are further preferably used in connection with a parsing engine, which may be provided as a module that interfaces with the libpcap library (e.g., Practical Extraction and Reporting Language (Perl)). The combination results in a dynamically configurable firewall that can parse and trace network protocol hacking patterns using the capturing and parsing engines.

[0024] Although the monitoring system 50 is preferably implemented using packet daemons 52 and is shown as implemented in a router 58, it may be provided in connection with other components of a network to thereby monitor data traffic. The monitoring system 50 of the present invention is preferably provided as a software and hardware adaptive firewall 54 addition to, for example, a switch router 58, which detects and denies data traffic with patterns that are in contrast to normal traffic patterns (i.e., exceed user defined configurable parameters), thereby preventing hacking attacks on networks. Depending upon the security requirements of the network, the present invention may be configured to detect different levels of attacks. The preferred packet daemon of the IDS 52 of the present invention uses the information it collects to issue firewall rules that make up the adaptive firewall functionality.

Thus, in contrast to the claimed feature, Ontiveros [0008] deals with an intrusion detection system (IDS) that uses a packet daemon that captures, sorts, and catalogs network traffic on a packet-by-packet basis. The packets are captured for inspection, parsing and tracing. In [0024] is discussed implementations of packet daemons 52, e.g., as a software and hardware adaptive firewall 54. In Ontiveros [0024] is discussed that the daemon “detects and denies data traffic with patterns that are in contrast to normal traffic patterns (i.e., exceed user defined configurable parameters), thereby preventing hacking attacks on networks.” However, none of these teachings suggest the feature of claim 1 of “a process executed on the aggregator device to detect anomalies in connection patterns.” Ontiveros deals with detection of traffic patterns, e.g., “hit-count” table, each time a data packet is received, a preferred algorithm as described herein creates a new reference index (if one does not already exist) or increments the existing reference (i.e., counting packets), but not the feature of “connection patterns.”

The examiner argues that Ontiveros teaches “a process executed on the aggregator device to aggregate detected anomalies into the network events (paragraph [0026].” Ontiveros discloses:

[0026] In the most preferred embodiment, six threads handle the various functions of the monitoring system 50. Specifically, the following threads are preferably provided: (1) Main Thread: initializes IDS data structures, activates the other threads, and waits for the other threads to complete their processes; (2) ADS connections thread: sends buffers to ADS, if ADS is present; (3) Packet Capture Thread: processes each packet, updates hit counts, queues lockout start commands to the per-second thread, extracts various fields, buffers the fields for transmission to an Anomaly Detection System (ADS), and notifies ADS connection thread to send buffers; (4) Per-second thread: runs each second, starts and stops lockout periods, and clears “hit” count table as configured; (5) Increment count thread: to determine a lock-out condition; and (6) Signal Catching Thread: re-reads configuration file, handles IDS 52 process cleanup and termination

Claim 1, however, includes a process “to aggregate detected anomalies into the network events with the anomalies that are detected including denial of service attack anomalies and scanning attack anomalies.” While Ontiveros discusses thwarting attacks, Ontiveros does not have any process that detects anomalies in connection patterns and determines whether the anomalies should be aggregated into events that can be associated with these types of attacks (thus minimizing false positives). Accordingly, claim 1 is allowable over Ontiveros.

Claim 2 distinguishes over Ontiveros at least because Ontiveros does not produce connection patterns derived from the connection table

Claim 3 serves to differentiate the preceding claims over Ontiveros, at least because Ontiveros only arguably mentions collection of statistical information on packets, as in claim 3, but does not therefore suggest the features of the base claim.

Claims 5 and 7 are allowable over Ontiveros at least for the reasons discussed in claim 1.

Claim 8 distinguishes over Ontiveros because the reference does not disclose the connection table whether at paragraphs [0040] and [0044] or elsewhere. Similar arguments apply to claims 9-11.

Claim 12 includes the feature of the connection table includes a plurality of connection sub-tables to track data at different time scales.

The examiner argues that this feature is disclosed at [0042]. Paragraph [0042] is reproduced below:

[0042] With respect specifically to cataloging, such process occurs only if the system's logging is enabled. If enabled, the cataloging function preferably creates a small ASCII file which provides information captured from the data packets, including for example source and destination MAC addresses and IP Addresses, packet type, packet size and destination port. This file is preferably transmitted using a secure channel on a short-time based interval to a large RDBMS.

However, in contrast to the claimed feature, paragraph [0042] only deals with how the ASCII file is transmitted, not the time basis or the feature of plural sub-tables as required by the claim.

Claim 13 further distinguishes because Ontiveros fails to disclose the connection sub-tables include a time slice connection table that operates on a small unit of time and at least one other sub table that operates on a larger unit of time than the time slice sub-table with each sub table holding the sum of records received from all collectors during respective units of time.

Claims 14-16, 18-22 and newly added claims 28-36 are allowable over Ontiveros for analogous reasons as given above.

The examiner rejected Claims 23-27 under 35 U.S.C. 102(b) as being anticipated by Venkatraman (5,923,849).

The examiner stated:

As per claim 23, Venkatraman discloses a method of detecting a new host connecting to a network comprises: receiving statistics collected from a host in the network and indicating to a console that the host is a new host if, during a period of time T, the host transmits at least N packets and receives at least N packets, and if the host had never transmitted and received more than N packets in any previous period of time with a duration of T (col. 12, lines 10-15 and 33-39) and (col. 14, lines 39-53).

Claim 23 is directed to a computer implemented method for detecting a new host connecting to a network. Claim 23 includes the features of receiving statistics collected from a host in the network and indicating to a console that the host is a new host if, during a period of time T, the host transmits at least N packets and receives at least N packets, and if the host had

never transmitted and received more than N packets in any previous period of time with a duration of T.

Venkatraman neither at (col. 12, lines 10-15 and 33-39) and (col. 14, lines 39-53), nor elsewhere suggests the combination of these features. Venkatraman is directed to techniques to thwart establishment of covert channels, e.g., using techniques that “involve the direct or indirect modification of storage memory by one process (the sender of a covert message) and the direct and indirect reading of the memory location by another process (the receiver of the covert message)” or “Covert timing channels ... when the sender process modulates the use of its own resources in a manner that affects the response of the receiver process.” None of the teachings in Venkatraman are directed to the algorithm claimed in claim 23 to detect a new host.

Claim 24

Claim 24 includes the features of “... determining ... if both a mean historical rate of server response packets from a host is greater than M and a ratio of a standard deviation of historical rate of server response packets from the host to a mean profiled rate of server response packets from the host is less than R over a period of time and indicating the host as a potential failed host if both conditions are present.

The examiner argues that:

As per claim 24, Venkatraman discloses a method of detecting a failed host in a network comprises: determining if both a mean historical rate of server response packets from a host is greater than M, and a ratio of a standard deviation of historical rate of server response packets from the host to a mean profiled rate (col. 7, lines 41-50) of server response packets from the host is less than R over a period of time (col. 10, lines 25-47); and indicating the host as a potential failed host if both conditions are present (col. 7, lines 26-30) and (col. 14, lines 39-53).

Applicant disagrees.

At col. 7, lines 41-50, Venkatraman discusses “... a fault dictionary” and “Comparing an observed out-of-normal communication pattern to known fault patterns in a fault dictionary.” At col. 10, lines 25-47 Venkatraman discusses “... an example of the distribution of the change in system communication traffic volume over one minute intervals” Venkatraman discusses “The normal sustained burst volume or baseline volume ... and the mean number of packets

transmitted per minute ...” Venkatraman also discusses that “Two standard deviations for the distribution shown in FIG. 8 is a 23.95 percentage change in traffic volume.” Venkatraman uses the threshold for auditing of the communication traffic to consider out of baseline and audited.”

However, the precise limitations of “determining ... if both a mean historical rate of server response packets from a host is greater than M and a ratio of a standard deviation of historical rate of server response packets from the host to a mean profiled rate of server response packets from the host is less than R over a period of time” are not suggested in any of these passages.

The examiner also argues that “indicating the host as a potential failed host if both conditions are present (col. 7, lines 26-30) and (col. 14, lines 39-53).” Col. 7, lines 26-30 merely establishes the conditions for an audit, whereas col. 14 lines 39-53 are directed to the possibility that an anomaly is due to a fault in the system. Neither of these passages however is directed to the claimed feature of “indicating the host as a potential failed host if both conditions are present.”

Claims 25-27 are allowable at least for the reasons discussed in claim 24. Newly added claim 37 is allowable for the reasons discussed for claim 23 and claims 38-41 are allowable for reasons discussed for claim 24.

It is believed that all the rejections and/or objections raised by the examiner have been addressed.

In view of the foregoing remarks, applicant respectfully submits that the application is in condition for allowance and such action is respectfully requested at the examiner's earliest convenience.

All of the dependent claims are patentable for at least the reasons for which the claims on which they depend are patentable.

Canceled claims, if any, have been canceled without prejudice or disclaimer.

Any circumstance in which the applicant has (a) addressed certain comments of the examiner does not mean that the applicant concedes other comments of the examiner, (b) made arguments for the patentability of some claims does not mean that there are not other good reasons for patentability of those claims and other claims, or (c) amended or canceled a claim

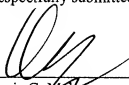
does not mean that the applicant concedes any of the examiner's positions with respect to that claim or other claims.

Please charge the Petition for Extension of Time fee of **\$60** and the excess claims fees of **\$540** to Deposit Account No. 06-1050. Please apply any other charges or credits to Deposit Account No. 06-1050.

Respectfully submitted,

Date: _____

5/6/08



Denis G. Maloney
Reg. No. 29,670

Fish & Richardson P.C.
225 Franklin Street
Boston, MA 02110
Telephone: (617) 542-5070
Facsimile: (617) 542-8906